

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平10-301772

(43) 公開日 平成10年(1998)11月13日

(51) Int.Cl.⁶

G 0 6 F 9/06

識別記号

5 5 0

F I

G 0 6 F 9/06

5 5 0 A

5 5 0 E

5 5 0 Z

審査請求 未請求 請求項の数14 O L (全 15 頁)

(21) 出願番号 特願平9-112179

(22) 出願日 平成9年(1997)4月30日

(71) 出願人 000002185

ソニー株式会社

東京都品川区北品川6丁目7番35号

(72) 発明者 竹内 彰一

東京都品川区北品川6丁目7番35号 ソニー株式会社内

(72) 発明者 難波 慎二

東京都品川区北品川6丁目7番35号 ソニー株式会社内

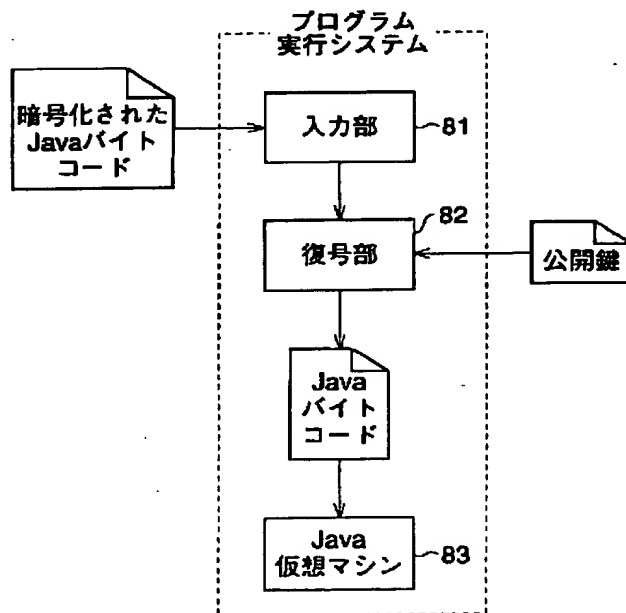
(74) 代理人 弁理士 稲本 義雄

(54) 【発明の名称】 情報処理装置および情報処理方法、並びに記録媒体

(57) 【要約】

【課題】 あるプログラム実行環境において、正当なソフトウェア開発者が開発したプログラムのみが実行されるようにし、その他のプログラムの実行を制限する。

【解決手段】 プログラム実行システムを構成する入力部81には、所定の秘密キーを用いて暗号化されたアプリケーションプログラムが入力され、その暗号化されたアプリケーションプログラムとしての暗号文は、そのまま、復号部82に供給される。復号部82には、入力部81から暗号文が供給される他、その暗号文を作成するのに用いた秘密キーに対応する公開キーも供給されるようになされており、そこでは、公開キーを用いて、暗号文が復号化され、その復号結果としてのJavaバイトコードが、Java仮想マシン83に供給される。Java仮想マシン83では、復号部82からのJavaバイトコードが解釈、実行される。



【特許請求の範囲】

【請求項1】 プログラムを実行するための処理を行う情報処理装置であって、

暗号化された前記プログラムを復号化する復号化手段と、

前記復号化手段が出力する前記プログラムを実行する実行手段とを備えることを特徴とする情報処理装置。

【請求項2】 プログラムを実行するための処理を行う情報処理方法であって、

暗号化された前記プログラムを復号化し、

その復号化されたプログラムを実行することを特徴とする情報処理方法。

【請求項3】 コンピュータに、暗号化されたプログラムを復号化させ、その復号化されたプログラムを実行させるためのプログラムが記録されていることを特徴とする記録媒体。

【請求項4】 暗号化されたプログラムの復号化が所定のキーを用いて行われる場合において、前記所定のキーも記録されていることを特徴とする請求項3に記載の記録媒体。

【請求項5】 プログラムを処理する情報処理装置であって、

請求項1に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化する暗号化手段を備えることを特徴とする情報処理装置。

【請求項6】 プログラムを処理する情報処理方法であって、

請求項1に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化することを特徴とする情報処理方法。

【請求項7】 請求項1に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化したものが記録されていることを特徴とする記録媒体。

【請求項8】 プログラムを実行するための処理を行う情報処理装置であって、

前記プログラムを実行する実行手段と、

前記プログラムが正当なものかどうかを確認する確認手段と、

前記確認手段により正当なものであることが確認された前記プログラムを、前記実行手段に供給する供給手段とを備えることを特徴とする情報処理装置。

【請求項9】 前記プログラムには、署名が付加されており、

前記確認手段は、前記署名に基づいて、前記プログラムが正当なものかどうかを確認することを特徴とする請求項8に記載の情報処理装置。

【請求項10】 プログラムを実行するための処理を行う情報処理方法であって、

前記プログラムが正当なものかどうかを確認し、

正当なものであることが確認された場合のみ、前記プログラムを実行することを特徴とする情報処理方法。

【請求項11】 コンピュータに、プログラムが正当なものかどうかを確認させ、正当なものであることが確認された場合のみ、前記プログラムを実行させるためのプログラムが記録されていることを特徴とする記録媒体。

【請求項12】 プログラムを処理する情報処理装置であって、

請求項8に記載の情報処理装置において正当なものであると確認されるように、プログラムを処理する処理手段を備えることを特徴とする情報処理装置。

【請求項13】 プログラムを処理する情報処理方法であって、

請求項8に記載の情報処理装置において正当なものであると確認されるように、プログラムを処理することを特徴とする情報処理方法。

【請求項14】 請求項8に記載の情報処理装置において正当なものであると確認されるように処理されたプログラムが記録されていることを特徴とする記録媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、情報処理装置および情報処理方法、並びに記録媒体に関し、特に、例えば、あるプログラム実行環境において、正当なソフトウェア開発者が開発したプログラムのみが実行されるようにする情報処理装置および情報処理方法、並びに記録媒体に関する。

【0002】

【従来の技術】最近急速に普及してきたインターネットに適していることから、Java（米国Sun Microsystems社の商標）が注目されている。Javaは、オブジェクト指向言語であるJava言語や、そのJava言語で記述されたプログラム（以下、適宜、Javaプログラムという）の実行に適したプロセッサのアーキテクチャを定義した仮想マシン（以下、適宜、Java仮想マシンという）、その他のJavaと関連する要素を呼ぶのに、あるいは、それらの総称として用いられる。なお、ここでいう仮想マシンとは、ユーザに対して1台のコンピュータを複数のコンピュータに仮想的に見せる場合の仮想マシンではなく、言語処理系を実装する場合に想定する仮想的なマシンを意味する。

【0003】Java仮想マシンは、種々のソフトウェアやOS（Operating System）、ハードウェアで動作するように実装される。一方、Javaプログラムは、Java仮想マシンの命令セットからなるバイナリコードにコンパイルされる。このバイナリコードは、Java仮想マシンが動作するどのようなハードウェアでも実行することができる。従って、Java仮想マシンさえ動作すれば、コンパイル済みのJavaプログラムは、種々のプラットフォームで実行することができる。

【0004】

【発明が解決しようとする課題】上述したように、Java仮想マシンを実装すれば、どのようなマシン上でも、Javaプログラムを実行することができることから、Java仮想マシンが、多数のユーザに普及することが予想され、さらに、そのような多数のユーザ向けに、多くのアプリケーションプログラムが開発、配布（有償、無償を問わない）されることが予想される。

【0005】このような状況の下、Java仮想マシンその他のプログラム実行環境を開発または配布した者からすれば、自身が開発等したプログラム実行環境下において実行される、第三者により開発されたアプリケーションプログラムの配布を制限したい場合がある。即ち、例えば、ライセンス契約を結んだ者だけに、アプリケーションプログラムの配布を許可したい場合がある。

【0006】一方、Java仮想マシンでは、JavaコンパイラでJavaプログラムを、バイトコード（Javaバイトコード）と呼ばれる中間コードにコンパイルしたものが解釈されて実行されるが、Javaバイトコードは、それを逆コンパイルすることにより、比較的容易に理解することができるため、リバースエンジニアリングを簡単にすることができる。従って、アプリケーションプログラムの、他人による模倣や改竄を防止する必要がある。

【0007】本発明は、このような状況に鑑みてなされたものであり、あるプログラム実行環境下におけるプログラムの実行を制限することができるようにし、さらに、プログラムの模倣や改竄を防止することができるようにもするものである。

【0008】

【課題を解決するための手段】請求項1に記載の情報処理装置は、暗号化されたプログラムを復号化する復号化手段と、復号化手段が出力するプログラムを実行する実行手段とを備えることを特徴とする。

【0009】請求項2に記載の情報処理方法は、暗号化されたプログラムを復号化し、その復号化されたプログラムを実行することを特徴とする。

【0010】請求項3に記載の記録媒体は、コンピュータに、暗号化されたプログラムを復号化させ、その復号化されたプログラムを実行させるためのプログラムが記録されていることを特徴とする。

【0011】請求項5に記載の情報処理装置は、請求項1に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化する暗号化手段を備えることを特徴とする。

【0012】請求項6に記載の情報処理方法は、請求項1に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化することを特徴とする。

【0013】請求項7に記載の記録媒体は、請求項1に

記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化したものが記録されていることを特徴とする。

【0014】請求項8に記載の情報処理装置は、プログラムが正当なものかどうかを確認する確認手段と、確認手段により正当なものであることが確認されたプログラムを、そのプログラムを実行する実行手段に供給する供給手段とを備えることを特徴とする。

【0015】請求項10に記載の情報処理方法は、プログラムが正当なものかどうかを確認し、正当なものであることが確認された場合のみ、プログラムを実行することを特徴とする。

【0016】請求項11に記載の記録媒体は、コンピュータに、プログラムが正当なものかどうかを確認させ、正当なものであることが確認された場合のみ、プログラムを実行させるためのプログラムが記録されていることを特徴とする。

【0017】請求項12に記載の情報処理装置は、請求項8に記載の情報処理装置において正当なものであると確認されるように、プログラムを処理する処理手段を備えることを特徴とする。

【0018】請求項13に記載の情報処理方法は、請求項8に記載の情報処理装置において正当なものであると確認されるように、プログラムを処理することを特徴とする。

【0019】請求項14に記載の記録媒体は、請求項8に記載の情報処理装置において正当なものであると確認されるように処理されたプログラムが記録されていることを特徴とする。

【0020】請求項1に記載の情報処理装置においては、復号化手段は、暗号化されたプログラムを復号化し、実行手段は、復号化手段が出力するプログラムを実行するようになされている。

【0021】請求項2に記載の情報処理方法においては、暗号化されたプログラムを復号化し、その復号化されたプログラムを実行するようになされている。

【0022】請求項3に記載の記録媒体には、コンピュータに、暗号化されたプログラムを復号化させ、その復号化されたプログラムを実行させるためのプログラムが記録されている。

【0023】請求項5に記載の情報処理装置においては、暗号化手段が、請求項1に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化するようになされている。

【0024】請求項6に記載の情報処理方法においては、請求項1に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化するようになされている。

【0025】請求項7に記載の記録媒体には、請求項1に記載の情報処理装置で実行可能なコードに復号化され

る暗号文に、プログラムを暗号化したものが記録されている。

【0026】請求項8に記載の情報処理装置においては、確認手段は、プログラムが正当なものかどうかを確認し、供給手段は、確認手段により正当なものであることが確認されたプログラムを、そのプログラムを実行する実行手段に供給するようになされている。

【0027】請求項10に記載の情報処理方法においては、プログラムが正当なものかどうかを確認し、正当なものであることが確認された場合のみ、プログラムを実行するようになされている。

【0028】請求項11に記載の記録媒体には、コンピュータに、プログラムが正当なものかどうかを確認させ、正当なものであることが確認された場合のみ、プログラムを実行させるためのプログラムが記録されている。

【0029】請求項12に記載の情報処理装置においては、処理手段が、請求項8に記載の情報処理装置において正当なものであると確認されるように、プログラムを処理するようになされている。

【0030】請求項13に記載の情報処理方法においては、請求項8に記載の情報処理装置において正当なものであると確認されるように、プログラムを処理するようになされている。

【0031】請求項14に記載の記録媒体には、請求項8に記載の情報処理装置において正当なものであると確認されるように処理されたプログラムが記録されている。

【0032】

【発明の実施の形態】以下に、本発明の実施の形態を説明するが、その前に、特許請求の範囲に記載の発明の各手段と以下の実施の形態との対応関係を明らかにするために、各手段の後の括弧内に、対応する実施の形態（但し、一例）を付加して、本発明の特徴を記述すると、次のようになる。

【0033】即ち、請求項1に記載の情報処理装置は、プログラムを実行するための処理を行う情報処理装置であって、暗号化されたプログラムを復号化する復号化手段（例えば、図12に示す復号部82など）と、復号化手段が出力するプログラムを実行する実行手段（例えば、図12に示すJava仮想マシン83など）とを備えることを特徴とする。

【0034】請求項5に記載の情報処理装置は、プログラムを処理する情報処理装置であって、請求項1に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化する暗号化手段（例えば、図10に示すプログラムの処理ステップS14など）を備えることを特徴とする。

【0035】請求項8に記載の情報処理装置は、プログラムを実行するための処理を行う情報処理装置であっ

て、プログラムを実行する実行手段（例えば、図16に示すJava仮想マシン83など）と、プログラムが正当なものかどうかを確認する確認手段（例えば、図16に示す署名確認部103など）と、確認手段により正当なものであることが確認されたプログラムを、実行手段に供給する供給手段（例えば、図16に示す仮想マシン入力制御部104など）とを備えることを特徴とする。

【0036】請求項12に記載の情報処理装置は、プログラムを処理する情報処理装置であって、請求項8に記載の情報処理装置において正当なものであると確認されるように、プログラムを処理する処理手段（例えば、図14に示すプログラムの処理ステップS24およびS25など）を備えることを特徴とする。

【0037】なお、勿論この記載は、各手段を上記したものに限定することを意味するものではない。

【0038】次に、本発明は、Javaのような仮想マシンの他、現実のマシンそのものについても適用可能であるが、ここでは、本発明を、Java仮想マシンに適用した場合を例に説明する。

【0039】なお、Javaについては、例えば、日経BP社発行の日経エレクトロニクスの1996.3.25（no.658）、同1996.6.17（no.664）などに、その詳細が記載されている。そこで、ここでは、簡単に説明する。

【0040】Java仮想マシンは、抽象化された実行機械であり、その実態は、実際の計算機で実行されるプログラムである。そして、Java仮想マシンは、実際の計算機と同様に、プログラムカウンタや、スタックレジスタ、汎用レジスタ、スタックやヒープとしてのメモリ、その他の資源を有するが、これらの資源は、実際の計算機の資源にマッピングされている。

【0041】即ち、図1に示すように、いま実際の計算機1が、中央演算処理装置2、その中央演算処理装置2が内蔵するレジスタ3、メモリ4などの資源を有するとして、この計算機1に、Java仮想マシン11を実装すると、そのJava仮想マシン11の資源として、実際の計算機1の資源がマッピングされる。図1の実施の形態では、Java仮想マシン11は、資源として、レジスタ13や、メモリ14などを有しており、レジスタ13はレジスタ3に、メモリ14の200番地は、メモリ4の100番地に、それぞれマッピングされている。

【0042】実際の計算機1では、中央演算処理装置2に対する命令は、その資源に対する操作として実行されるが、Java仮想マシン11においても、その資源に対する操作として実行される命令が定義されている。このJava仮想マシン11に対する命令を記述するための言語がJava言語であり、Java仮想マシン11では、このJava言語で記述されたソースプログラムを、Javaコンパイラで、Javaバイトコードにコンパイルしたものが、解釈、実行される。

【0043】即ち、図2に示すように、Java言語で記述されたソースプログラムであるJava言語プログラムは、Javaコンパイラ21でコンパイルされ、Javaバイトコードとされる。このJavaバイトコードが、Java仮想マシン11に入力され、Java仮想マシン11では、Javaバイトコードが、実際の計算機1（中央演算処理装置2）が解釈することのできる機械語コード（マシン語）に変換される。具体的には、例えば、図3に示すように、「move #125、レジスタ13」というJavaバイトコードで記述された「数字の125を、レジスタ13にセット」という命令（Javaバイトコード命令）が、Java仮想マシン11に入力された場合、Java仮想マシン11では、このJavaバイトコードが、「move #125、レジスタ3」という機械語コードで記述された命令（マシン語命令）に変換される。

【0044】そして、計算機1では、この機械語コードで記述された命令が実行されることにより、図4（A）に示すように、計算機1のレジスタ3に、数字の125がセットされる。

【0045】上述したように、計算機1のレジスタ3には、Java仮想マシン11のレジスタ13がマッピングされており、従って、図4（A）に示したように、計算機1のレジスタ3に、数字の125をセットすることは、Java仮想マシン11から見れば、図4（B）に示すように、そのレジスタ13に数字の125をセットすることになる。

【0046】以上のように、Java仮想マシン11へのJavaバイトコードによる命令は、計算機1用の機械語コードに変換され、Java仮想マシン11の資源にマッピングされている計算機1の資源に対する操作として実行される。この操作は、Java仮想マシン11から見れば、その資源（Java仮想マシン11の資源）に対する操作となり、Javaバイトコードによる命令が実行されたことになる。

【0047】従って、前述したように、Java仮想マシンを、実際の計算機（コンピュータ）に実装することで、その計算機が使用しているCPU（Central Processing Unit）やOSに無関係に、コンパイル済みのJavaプログラムを実行することができる。

【0048】ここで、Javaバイトコードを機械語コードに変換して実行する手法としては、例えば、Basic言語によるプログラムを実行する場合に採用されている、各命令を機械語コードに逐次翻訳して実行するインタープリタ形式と、命令を一括して機械語コードに翻訳して実行するJIT（Just In Time）コンパイラ形式などがある。

【0049】なお、Basic言語によるプログラムを実行する場合に採用されているインタープリタ形式は、そのソースコードを翻訳する点で、中間コードであるJ

avaバイトコードを翻訳する場合と異なるが、ここでは、これらを特に区別しない（区別する必要もない）。

【0050】次に、図5は、本発明を適用したプログラム提供システム（システムとは、複数の装置が論理的に集合したものをいい、各構成の装置が同一筐体中にあるか否かは問わない）の一実施の形態の構成例を示している。

【0051】このプログラム提供システムにおいては、ソフトウェア開発者が、プログラム認証機関により認証されていないアプリケーションプログラムをユーザに配布した場合に、そのユーザ端末33上での、そのアプリケーションプログラムの実行を制限するようになされている。

【0052】即ち、ソフトウェア開発者は、例えば、Java仮想マシン上で動作するアプリケーションプログラムを開発すると、それをコンパイルしてJavaバイトコードとしたものを、そのソフトウェア開発者サーバ31から、例えば、インターネットや、公衆回線、CATV網、地上波、衛星回線、その他でなるネットワーク34を介して、プログラム認証機関サーバ32に送信する。

【0053】プログラム認証機関サーバ32では、ソフトウェア開発者サーバ31からJavaバイトコードを受信すると、それを認証し、その認証したJavaバイトコードを、ネットワーク34を介して、ソフトウェア開発者サーバ31に送信する。ソフトウェア開発者サーバ31では、プログラム認証機関サーバ32からの認証されたJavaバイトコードであるアプリケーションプログラムが受信され、記憶される。

【0054】そして、ソフトウェア開発者サーバ31においては、ユーザ端末33からアプリケーションプログラムの要求があると、そのアプリケーションプログラムを、ネットワーク34を介して、ユーザ端末33に送信する。ユーザ端末33には、プログラム認証機関、またはプログラム認証機関にプログラムの認証を依頼している者が開発または配布したプログラム実行環境としての、例えば、Java仮想マシンが実装されている。そして、Java仮想マシンとしてのユーザ端末33では、ソフトウェア開発者サーバ31から受信したアプリケーションプログラムが、プログラム認証機関によって認証されているものであるときのみ、そのアプリケーションプログラムが、正常に実行される。

【0055】即ち、ソフトウェア開発者サーバ31から受信したアプリケーションプログラムが、プログラム認証機関によって認証されていない場合には、Java仮想マシンとしてのユーザ端末33では、そのアプリケーションプログラムは実行されない（実行されても、正常には実行されない）。

【0056】従って、結果として、プログラム実行環境としてのJava仮想マシンを開発等した者は、そのJ

Java仮想マシンにおいて実行される、第三者により開発されたアプリケーションプログラムの、いわば勝手な配布を制限し、例えば、ライセンス契約を結んだソフトウェア開発者だけに、アプリケーションプログラムの配布を許可することができる。

【0057】なお、ソフトウェア開発者が、アプリケーションプログラムを、例えば、CD (Compact Disc) - ROM、磁気ディスク、その他でなる記録媒体35に記録して、郵送、店頭における販売、その他の手段で、ユーザに配布する場合においても、上述の場合と同様に、そのアプリケーションプログラムが、プログラム認証機関によって認証されていないときには、Java仮想マシンとしてのユーザ端末33では、そのアプリケーションプログラムは実行されない。

【0058】また、上述の場合においては、ソフトウェア開発者とプログラム認証機関との間で、ネットワーク34を介して、データのやりとりを行うようにしたが、これらの間でのデータのやりとりは、そのデータを記録した記録媒体を郵送などすることによって行うことも可能である。

【0059】さらに、図5の実施の形態では、ソフトウェア開発者サーバ31、プログラム認証機関サーバ32、またはユーザ端末33を、それぞれ1つしか設けていないが、これらは、複数設けることが可能である。

【0060】次に、図6は、図5のソフトウェア開発者サーバ31の構成例を示している。

【0061】CPU41は、補助記憶装置46に記憶（記録）されたオペレーティングシステムの制御の下、同じく補助記憶装置46に記憶されたプログラムを実行することで、各種の処理を行う。ROM (Read Only Memory) 42は、例えば、IPL (Initial Program Loading) のプログラムなどを記憶している。RAM (Random Access Memory) 43は、CPU41が実行するプログラムや、CPU1の動作上必要なデータを記憶する。入力部44は、例えば、キーボードやマウスなどで構成され、所定のデータやコマンドを入力するときなどに操作される。出力部45は、ディスプレイやプリンタなどで構成され、所定の情報を表示、印刷する。補助記憶装置46は、例えば、ハードディスクなどで構成され、オペレーティングシステム、その他のCPU41が実行するプログラムなどを記憶している。さらに、補助記憶装置46は、CPU41の処理結果その他の必要なデータなども記憶する。通信制御部47は、ネットワーク34を介して行われる通信を制御する。

【0062】次に、図7は、図5のプログラム認証機関サーバ32の構成例を、図8は、図5のユーザ端末33の構成例を、それぞれ示している。

【0063】プログラム認証機関サーバ32は、CPU51乃至通信制御部57で構成され、また、ユーザ端末33は、CPU61乃至通信制御部67で構成され、こ

れらは、図6におけるCPU41乃至通信制御部47とそれぞれ同様に構成されるので、その説明は省略する。

【0064】次に、図9のフローチャートを参照して、ソフトウェア開発者サーバ31の処理について説明する。

【0065】ソフトウェア開発者が開発した、Java仮想マシン上で実行されるアプリケーションプログラムは、例えば、補助記憶装置46に記録（記憶）される。そして、ソフトウェア開発者サーバ31では、まず最初に、ステップS1において、CPU41が、補助記憶装置46に記憶されたアプリケーションプログラムを、Javaコンパイラのプログラムにしたがってコンパイルし、Javaバイトコードとする。このJavaバイトコードは、再び、補助記憶装置46に供給されて記憶される。

【0066】そして、ステップS2に進み、通信制御部47は、補助記憶装置46に記憶されたアプリケーションプログラムとしてのJavaバイトコードを読み出し、ネットワーク34を介して、プログラム認証機関サーバ32に送信し、ステップS3に進む。ステップS3では、CPU41において、認証されたアプリケーションプログラム（Javaバイトコード）としての、そのアプリケーションプログラムを暗号化した暗号文が、プログラム認証機関サーバ32から送信されてきたかどうか判定され、まだ、送信されてきていないと判定された場合、ステップS3に戻る。

【0067】また、ステップS3において、暗号文が送信されてきたと判定された場合、ステップS4に進み、通信制御部47において、その暗号文が受信され、ステップS5に進む。ステップS5では、通信制御部47で受信された暗号文が、補助記憶装置46に転送されて記憶され、処理を終了する。

【0068】次に、図10のフローチャートを参照して、プログラム認証機関サーバ32の処理について説明する。

【0069】プログラム認証機関は、例えば、プログラム実行環境としてのJava仮想マシンを開発または配布した者、あるいは、その者の依頼を受けた機関で、プログラム認証機関サーバ32では、例えば、ライセンス契約を結んだ者からのアプリケーションプログラムを認証するプログラム認証処理を行うようになされている。

【0070】即ち、プログラム認証機関サーバ32のCPU51では、まず最初に、ステップS11において、例えば、ソフトウェア開発者サーバ31などから、認証対象のアプリケーションプログラムとしてのJavaバイトコードが、ネットワーク34を介して送信されてきたかどうか判定され、送信されてきていないと判定された場合、ステップS11に戻る。また、ステップS11において、Javaバイトコードが送信されてきたと判定された場合、ステップS12に進み、そのJava

バイトコードが、例えば、ライセンス契約を結んだソフトウェア開発者（以下、適宜、正規のソフトウェア開発者という）からのものであるかどうか、CPU51によって判定される。

【0071】即ち、プログラム認証機関は、Java仮想マシン上で実行されるアプリケーションプログラムの開発、販売等を許可することについてのライセンス契約を、ソフトウェア開発者との間で結ぶと、そのソフトウェア開発者に対して、例えば、IDおよびパスワードを発行する。そして、このようなライセンス契約を結んだ正規のソフトウェア開発者からは、認証対象のJavaバイトコードとともに、ライセンス契約時に発行されたIDおよびパスワードが送信されるようになされており、ステップS12では、このIDおよびパスワードに基づいて、Javaバイトコードが、正規のソフトウェア開発者からのものであるかどうか判定される。

【0072】ステップS12において、Javaバイトコードが、正規のソフトウェア開発者からのものでないと判定された場合、即ち、ライセンス契約を結んでいないソフトウェア開発者からJavaバイトコードが送信されてきた場合、ステップS13に進み、通信制御部57において、そのソフトウェア開発者に対して、ライセンス契約を結ばなければ、Javaバイトコードを認証することができない旨のメッセージが送信され、処理を終了する。

【0073】一方、ステップS12において、Javaバイトコードが、正規のソフトウェア開発者からのものであると判定された場合、ステップS14に進み、CPU51において、そのJavaバイトコードが暗号化され、これにより暗号文とされることで、その認証が行われる。

【0074】そして、ステップS15に進み、通信制御部57において、Javaバイトコードの認証結果としての、その暗号文が、Javaバイトコードを送信してきたソフトウェア開発者、即ち、ここでは、例えば、ソフトウェア開発者サーバ31に、ネットワーク34を介して送信され、処理を終了する。

【0075】次に、プログラム認証機関サーバ32において行われる、ステップS14の暗号化の方法について説明する。

【0076】プログラム認証機関サーバ32では、Javaバイトコードの暗号化が、例えば、RSA方式（MITの3人の研究者により考案されたもので、RSAは、その3人の研究者の頭文字をとったものである）などに代表される公開鍵暗号化方式によって行われるようになされている。

【0077】即ち、図11は、公開鍵暗号化方式による暗号化／復号化システムの構成例を示している。

【0078】暗号化器71には、暗号化対象である平文が入力される。そして、暗号化器71では、秘密キーと

呼ばれる各個人に特有の暗号キーを用いて、平文が暗号化され、暗号文とされる。

【0079】一方、復号化器72には、暗号化器71で暗号化された暗号文が入力される。そして、復号化器72では、公開キーと呼ばれる広く一般に公開されている復号キーを用いて、暗号文が復号化され、元の平文とされる。

【0080】プログラム認証機関サーバ32では、ソフトウェア開発者サーバ31からのJavaバイトコードが、それに特有の秘密キーを用いて暗号化されて、暗号文とされる。

【0081】なお、暗号化の手法は、公開鍵暗号化方式に限定されるものではなく、その他、例えば、DES（Data Encryption Standard）方式（IBM社により開発され、米国連邦政府の標準として実用化されたもの）などに代表される秘密鍵暗号方式その他の方式を採用することが可能である。

【0082】次に、図12は、ユーザ端末33においてアプリケーションプログラムを実行するプログラム実行環境としてのプログラム実行システムの機能的構成例を示している。

【0083】入力部81は、暗号文（暗号化されたJavaバイトコード）を受け付け、復号部82に供給するようになされている。復号部82は、例えば、図11の復号化器72としての処理を行うもので、入力部81の出力を、公開キーを用いて復号化し、元のJavaバイトコードとするようになされている。復号部82で得られたJavaバイトコードは、Java仮想マシン83に供給されるようになされており、Java仮想マシン83は、復号部82からのJavaバイトコードにしたがった処理を実行するようになされている。

【0084】以上のように構成されるプログラム実行システムでは、まず最初に、入力部81において、アプリケーションプログラムとしての暗号文が取得される。即ち、例えば、あらかじめ、ソフトウェア開発者サーバ31から、ネットワーク34を介して、アプリケーションプログラムとしての暗号文を受信し、その暗号文が、ファイルとして、ユーザ端末33の補助記憶装置67に記憶されている場合や、アプリケーションプログラムとしての暗号文がファイルとして記録された記録媒体35がユーザ端末33にセットされる場合などには、入力部81は、そのファイルをオープンし、暗号文を読み出す。

【0085】また、例えば、ソフトウェア開発者サーバ31が、インターネットとしてのネットワーク34に接続されており、そのようなソフトウェア開発者サーバ31において、アプリケーションプログラムとしての暗号文が、URL（Uniform Resource Locator）と対応付けられている場合に、ユーザが、入力部64を操作して、そのURLを指定したときには、入力部81は、ソフトウェア開発者サーバ31からネットワーク34を介して

送信されてくるアプリケーションプログラムとしての暗号文を受信する。

【0086】さらに、例えば、ソフトウェア開発者サーバ31が、地上波や衛星回線としてのネットワーク34を介して、アプリケーションプログラムとしての暗号文をデジタル放送などしている場合には、入力部81は、その放送されてくる暗号文を受信する。

【0087】入力部81は、以上のようにして取得したアプリケーションプログラムとしての暗号文を、そのまま、復号部82に供給する。

【0088】復号部82には、入力部81から暗号文が供給される他、その暗号文を作成するのに用いた秘密キーに対応する公開キーも供給されるようになされている。

【0089】ここで、公開キーは、例えば、プログラム認証機関が管理しており、ユーザからの要求に応じて、ネットワーク34を介して、ユーザ端末33に送信される。あるいは、公開キーは、郵送などによって、ユーザに配送される。また、プログラムとしての暗号文がファイルとして記録された記録媒体35がユーザに配布される場合には、公開キーは、記録媒体35に、暗号文とともに記録しておくようにすることもできる。さらに、公開キーは、プログラム認証機関サーバ32におけるJavaバイトコードの認証の後に、その認証結果としての暗号文とともに、ソフトウェア開発者サーバ31に送信するようにし、ソフトウェア開発者から、ユーザに配布するようにすることなども可能である。

【0090】復号部82では、公開キーを用いて、入力部81からの暗号文が復号化され、その復号結果としてのJavaバイトコードが、Java仮想マシン83に供給される。Java仮想マシン83では、復号部82からのJavaバイトコードが解釈、実行される。

【0091】以上のように、復号部82において、プログラム認証機関サーバ32での暗号化に用いられた秘密キーに対応する公開キー（秘密キーと対になる公開キー）を用いて、暗号文の復号化が行われ、その復号結果が、Java仮想マシン83に入力される。従って、復号部82に対して、プログラム認証機関で認証されていないJavaバイトコード、即ち、例えば、暗号化されていないJavaバイトコードや、プログラム認証機関サーバ32における暗号化アルゴリズムと異なるアルゴリズムで暗号化されたJavaバイトコード、あるいはそれと同一のアルゴリズムで暗号化されていたとしても、本来使用すべき秘密キーを用いずに暗号化されたJavaバイトコードなどが入力された場合においては、復号部82からは、Java仮想マシン83が正常に実行可能なJavaバイトコードは出力されないから（Java仮想マシン83が正常に実行可能なJavaバイトコードが、偶然に出力されることは有り得ないことではないが、ほとんどないに等しい）、結果として、Ja

va仮想マシン83上で動作するJavaバイトコードであって、プログラム認証機関で認証されていないものの、Java仮想マシン83が実装されたユーザ端末33を有するユーザへの配布を制限することができる。

【0092】即ち、Java仮想マシン83上で動作するJavaバイトコードの、そのJava仮想マシン83が実装されたユーザ端末33を有するユーザへの配布は、プログラム認証機関とライセンス契約を結んだソフトウェア開発者にのみ許可することができ、Java仮想マシン83の開発者や配布者は、Java仮想マシン83上で動作するJavaバイトコードの配布を希望するソフトウェア開発者から、いわば、Java仮想マシン83を使用したアプリケーションプログラムを配布するためのライセンス料を得ることが可能となる。

【0093】なお、Java仮想マシン83に対するJavaバイトコードの入力は、復号部82からのみ行うことができるようにしておく必要がある。

【0094】ここで、図12における復号部82では、何らかの入力があると、その入力に対して復号化処理が施され、その処理結果が出力される。従って、プログラム認証機関で認証されていないJavaバイトコードが復号部82に入力され、その処理結果が、Java仮想マシン83に供給されると、Java仮想マシン83は、通常暴走する。そこで、復号部82の出力が、正当（正常）なJavaバイトコードかどうかを確認し、正当なJavaバイトコードである場合にのみ、そのJavaバイトコードを、Java仮想マシン83に解釈、実行させるようにすることが可能である。即ち、Javaバイトコードには、その先頭に、マジック（magic）と呼ばれる32ビットのデータが配置されており、これが本来の値（16進数で、CAFEBABE）である場合にのみ、復号部82の出力が正当なJavaバイトコードであるとして、Java仮想マシン83に解釈、実行させるようにすることができる。この場合、Java仮想マシン83の暴走を防止することができる。

【0095】なお、プログラム認証機関サーバ32において、Javaバイトコード全体ではなく、例えば、上述のマジックなどの、Javaバイトコードの一部を暗号化することにより、Javaバイトコードの認証を行うことも可能であるが、マジックは32ビットという少ないビット数で構成されるため、復号部82において本来の値が出力されるような改竄を、Javaバイトコード全体を暗号化する場合に比較して容易に行うことができることが予想される。従って、暗号化は、Javaバイトコード全体に対して施すのが望ましい。

【0096】ところで、図12の復号部82における復号化アルゴリズムや、そこで用いるべき公開キーが、第三者に知られても、Java仮想マシン83上におけるアプリケーションプログラムの実行の制限の観点からは、暗号化アルゴリズムやその暗号化に用いる秘密キー

さえ知らなければ、特に問題はない。即ち、暗号文の復号化方法が知られても、正常に実行可能なJavaバイトコードをJava仮想マシン83に供給するために復号部82に与えるべき暗号文の作成方法を知らなければ、仮想マシン83上におけるアプリケーションプログラムの実行を制限することができる。

【0097】しかしながら、暗号文の復号化方法が知られた場合、その暗号文（プログラム認証機関においてJavaバイトコードを認証した結果としての暗号文）から、Javaバイトコードを得ることができる。Javaバイトコードは、前述したように、それを逆コンパイルすることにより、その内容を比較的容易に理解することができるため、リバースエンジニアリングを簡単に行うことができる。

【0098】そこで、このようなリバースエンジニアリングを防止すべく、暗号文の復号化方法は秘密にすることができる。即ち、例えば、暗号文の復号化に用いられる公開キーは、上述したように、一般に公開されるものであるが、これを秘密にすることなどができる。

【0099】図13は、公開キーを秘密にするようにしたプログラム実行システムの構成例を示している。なお、図中、図12における場合と対応する部分については、同一の符号を付してあり、以下では、その説明は、適宜省略する。

【0100】この実施の形態においては、例えば、Java仮想マシン83を含むプログラム実行システムを構成するプログラムの中の1カ所に、または複数箇所に分散して、公開キーが配置されており、復号部82では、その公開キーを用いて、暗号文の復号化が行われる。従って、この場合、公開キーは、プログラム実行システムの外部に漏れることがなく、その結果、暗号文が、不正に復号され、リバースエンジニアリングされることを防止すること（リバースエンジニアリングされる確率を低減すること）が可能となる。

【0101】次に、以上においては、ユーザ端末33において、基本的には、復号部82における復号化結果を、そのままJava仮想マシン83に入力するようにしたが、ユーザ端末33においては、Javaバイトコードが、プログラム認証機関で認証されたものであり、かつ改竄などが行われていない、いわば正当なものであるかどうかを確認し、その確認がなされたJavaバイトコードのみを、Java仮想マシン83に入力するようにすることなども可能である。

【0102】この場合、プログラム認証機関サーバ32では、ソフトウェア開発者サーバ31から送信されてくるアプリケーションプログラムとしてのJavaバイトコードに対して、例えば、図14に示すフローチャートにしたがったプログラム認証処理が行われる。

【0103】即ち、ステップS21乃至S23においては、図10のステップS11乃至S13における場合と

それぞれ同様の処理が行われる。

【0104】そして、ステップS22において、送信されてきたJavaバイトコードが、正規のソフトウェア開発者からのものであると判定された場合、ステップS24乃至S26に順次進み、そのJavaバイトコードに対して、それが正当なものであることを証明する署名（デジタルサイン）が付される。

【0105】即ち、ステップS24では、CPU51において、Javaバイトコードのダイジェストが作成され、ステップS25に進む。ステップS25では、CPU51において、ステップS24で作成されたダイジェストから、デジタルサインが作成される。そして、ステップS26に進み、通信制御部47において、そのデジタルサインが、Javaバイトコードに付され（このようにデジタルサイン（署名）が付されたJavaバイトコードを、以下、適宜、署名付きバイトコードという）、ソフトウェア開発者サーバ32に送信されて、処理を終了する。

【0106】次に、プログラム認証機関サーバ32において行われる、ステップS24乃至26の署名付きバイトコードの作成方法について説明する。

【0107】プログラム認証機関サーバ32では、署名付きバイトコードの作成が、例えば、RSA方式などに代表される公開鍵暗号化方式によって行われるようになっている。

【0108】即ち、図15は、公開鍵暗号化方式による、デジタルサイン（署名）を用いた暗号化／復号化システムの構成例を示している。

【0109】ダイジェスト作成器91には、認証対象である平文が入力される。そして、ダイジェスト作成器91では、入力された平文のダイジェストが、例えば、MD5やSHA-1などのアルゴリズムにしたがって作成される。

【0110】ここで、ダイジェストは、平文の機械的な凝縮文に相当し、入力としての平文が異なれば、そのダイジェストも異なるものが作成される。ダイジェストの作成は、平文を、例えば、ハッシュ関数を用いて変換することで行われる。

【0111】なお、データベースの検索を行うのに用いるキーワードがとり得る範囲の集合を、ある限られた数値範囲（レコード番号や配列の添字などに対応する）に写像する方法はハッシング（hashing）と呼ばれるが、この写像を行う変換関数がハッシュ関数である。

【0112】ダイジェスト作成器91で作成されたダイジェストは、暗号化器92に供給される。暗号化器92では、例えば、図11の暗号化器71における場合と同様に、ダイジェストが、秘密キーを用いて暗号化され、この暗号化されたダイジェストがデジタルサインとして出力される。そして、デジタルサインが、元の平文に付加され、署名付き平文として出力される。

【0113】一方、復号化器93には、署名付き平文を構成するデジタルサインが、また、ダイジェスト作成器94には、その残りの平文が、それぞれ入力される。復号化器93では、例えば、図11の復号化器72における場合と同様にして、デジタルサインが、公開キーを用いて復号化され、ダイジェストとされる。このダイジェストは、署名確認器95に供給される。

【0114】ダイジェスト作成器94では、ダイジェスト作成器91における場合と同様にして、そこに入力される平文のダイジェストが作成され、署名確認器95に供給される。

【0115】署名確認器95では、署名（デジタルサイン）の正当性が判断される（署名の確認が行われる）。即ち、署名確認器95では、復号化器93が出力するダイジェストが、ダイジェスト作成器94が出力するダイジェストと一致するかどうかを確認される。復号化器93が出力するダイジェストが、ダイジェスト作成器94が出力するダイジェストと一致しない場合、例えば、平文の改竄が行われたとして、あるいは、復号化器93で用いられた公開キーが正しくないとして、署名の正当性が否定される。

【0116】一方、復号化器93が出力するダイジェストが、ダイジェスト作成器94が出力するダイジェストと一致する場合、平文の改竄が行われておらず、かつ復号化器93で用いられた公開キーが正しいとして、署名の正当性が確認される。

【0117】署名確認器95には、署名付き平文を構成する平文も供給されるようになされており、そこでは、署名の正当性が確認されると、その平文が出力される。

【0118】プログラム認証機関サーバ32では、ソフトウェア開発者サーバ31からのJavaバイトコードが、上述の署名付き平文に相当する署名付きバイトコードとされることにより、そのJavaバイトコードが認証される。

【0119】なお、署名の手法は、上述した公開鍵暗号化方式に限定されるものではない。

【0120】次に、図16は、ユーザ端末33においてアプリケーションプログラムの正当性を確認し、正当なもののみを実行するプログラム実行環境としてのプログラム実行システムの機能的構成例を示している。なお、図中、図12における場合と対応する部分については、同一の符号を付してあり、以下では、その説明は、適宜省略する。

【0121】入力部101は、基本的には、図12における入力部81と同様に、そこへの入力を受け付けるようになされている。但し、入力部101には、署名付きバイトコード（署名（デジタルサイン）が付されたJavaバイトコード）が入力されるようになされており、そこでは、署名付きバイトコードが、署名とJavaバイトコードとに分離されて出力されるようになされ

ている。署名は、署名確認部103に、Javaバイトコードは、メッセージダイジェストシステム102および仮想マシン入力制御部104に、それぞれ供給されるようになされている。

【0122】メッセージダイジェストシステム102は、図15のダイジェスト作成器94と同様の処理を行うもので、入力部101からのJavaバイトコードからダイジェストを作成し、署名確認部103に供給するようになされている。署名確認部103は、図15の復号化器93および署名確認器95に相当するもので、入力部101からの署名の正当性を確認するようになされている。

【0123】即ち、署名確認部103には、入力部101から署名が、メッセージダイジェストシステム102からダイジェストが、それぞれ供給される他、例えば、図12の復号部82に公開キーが供給されるのと同様にして、署名を作成するときに用いた秘密キーに対応する公開キーが供給されるようになされている。そして、署名確認部103は、その公開キーを用いて、署名を復号化することにより、ダイジェストとし、そのダイジェストと、メッセージダイジェストシステム102からのダイジェストとを比較することで、署名の正当性を確認するようになされている。さらに、署名確認部103は、その確認結果に対応して、仮想マシン入力制御部104を制御するようになされている。

【0124】仮想マシン入力制御部104は、署名確認部103の制御にしたがって、入力部101からのJavaバイトコードの、Java仮想マシン83への供給を制御するようになされている。

【0125】以上のように構成されるプログラム実行システムでは、まず最初に、入力部101において、図12の入力部81における場合と同様にして、アプリケーションプログラムとしての署名付きバイトコードが取得される。そして、入力部101は、その署名付きバイトコードを、署名とJavaバイトコードとに分離し、署名を、署名確認部103に、Javaバイトコードを、メッセージダイジェストシステム102および仮想マシン入力制御部104に、それぞれ供給する。

【0126】メッセージダイジェストシステム102では、入力部101からのJavaバイトコードからダイジェストが作成され、署名確認部103に供給される。署名確認部103では、公開キーを用いて、入力部101からの署名が復号化されてダイジェストとされる。さらに、署名確認部103では、その復号化したダイジェストが、メッセージダイジェストシステム102からのダイジェストと比較され、それが一致するかどうかで、入力部101からの署名の正当性が確認される。

【0127】署名の正当性が確認された場合、即ち、署名を復号化したダイジェストが、メッセージダイジェストシステム102からのダイジェストと一致する場合、

署名確認部103は、入力部101からのJavaバイトコードを、Java仮想マシン83に出力するように、仮想マシン入力制御部104を制御する。この場合、仮想マシン入力制御部104は、署名確認部103の制御にしたがい、入力部101からのJavaバイトコードを、Java仮想マシン83に供給する。

【0128】従って、この場合、Java仮想マシン83では、入力部101から仮想マシン入力制御部104を介して供給されるJavaバイトコードが解釈、実行される。

【0129】一方、署名の正当性が確認されなかった場合、即ち、署名を復号化したダイジェストが、メッセージダイジェストシステム102からのダイジェストと一致しない場合、署名確認部103は、入力部101からのJavaバイトコードを、Java仮想マシン83に出力しないように、仮想マシン入力制御部104を制御する。

【0130】この場合、仮想マシン入力制御部104からJava仮想マシン83に対しては、入力部101からのJavaバイトコードは出力されず、従って、Java仮想マシン83では、特に処理は行われない。

【0131】以上から、アプリケーションプログラムの認証として署名を付す場合も、Java仮想マシン83上で動作するJavaバイトコードであって、プログラム認証機関で認証されていないものの、Java仮想マシン83が実装されたユーザ端末33を有するユーザへの配布を制限することができる。即ち、Java仮想マシン83上で動作するJavaバイトコードの、そのJava仮想マシン83が実装されたユーザ端末33を有するユーザへの配布は、プログラム認証機関とライセンス契約を結んだソフトウェア開発者にのみ許可することができ、Java仮想マシン83の開発者等は、Java仮想マシン83上で動作するJavaバイトコードの配布を希望するソフトウェア開発者から、ライセンス料を得ることが可能となる。

【0132】また、署名を付す場合においては、その署名を付したJavaバイトコードを改竄したものの、仮想マシン83上での実行も制限することができる。

【0133】なお、図16の実施の形態では、Java仮想マシン83に対するJavaバイトコードの入力は、仮想マシン入力制御部104からのみ行うことができるようにしておく必要がある。

【0134】ここで、Javaバイトコードに署名を付す場合においては、Javaバイトコードを暗号化する場合と異なり、Javaバイトコードそのものが存在する。従って、プログラム実行システムが署名の正当性を確認しないもの（例えば、入力部101の出力であるJavaバイトコードが、Java仮想マシン83に直接入力されるような構成のもの）であれば、そのプログラム実行システムでは、何の制限もなく、Javaバイト

コードを解釈して実行することができる。

【0135】即ち、逆にいえば、Javaバイトコードに署名を付す場合においては、その実行を制限したいJava仮想マシンの開発者や販売者、さらには、ユーザ端末33にJava仮想マシンを実装して販売する者などは、そのプログラム実行システムを、図16に示したように構成すれば良いし、その実行の制限を特に希望しない者は、プログラム実行システムを署名の正当性を確認しないような構成とすれば良い。

【0136】なお、本発明は、上述したインタープリタ形式およびJITコンパイラ形式のいずれのJava仮想マシンにも適用可能であるし、また、Java仮想マシン以外の仮想マシン、さらには、例えば、C言語や、C++言語などの処理系のようにプログラム実行システムへの入力が機械語コードの場合や、Basic言語の処理系のようにプログラム実行システムへの入力がソースコードの場合などについても適用可能である。

【0137】また、図12や図13、図16の実施の形態においては、プログラム実行システムを1つしか設けていないが、ユーザ端末33には、このプログラム実行システムを複数設けることも可能である。例えば、入力部81や101を複数設けた場合においては、複数経路からの暗号文や署名付きバイトコードの入力が可能となる。さらに、例えば、復号部82を複数設けた場合においては、複数の復号アルゴリズムにしたがって暗号文を復号することが可能となる。また、例えば、Java仮想マシン83を複数設けた場合には、複数のJavaバイトコード形式をサポートすることが可能となる。さらに、例えば、メッセージダイジェストシステム102および署名確認部103を複数設けた場合には、複数の手法それぞれで付された複数の署名の確認を行うことが可能となる。

【0138】なお、本実施の形態におけるJavaバイトコードとは、例えば、Java Application, Java Applet, Java Beans, Java Class Libraryその他の多数の形態のJavaバイトコードをすべて含むものである。

【0139】

【発明の効果】請求項1に記載の情報処理装置および請求項2に記載の情報処理方法によれば、暗号化されたプログラムが復号化され、その復号化されたプログラムが実行される。また、請求項3に記載の記録媒体には、コンピュータに、暗号化されたプログラムを復号化させ、その復号化されたプログラムを実行させるためのプログラムが記録されている。従って、暗号化されたプログラムのみが実行されるようにすることが可能となる。

【0140】請求項5に記載の情報処理装置および請求項6に記載の情報処理方法によれば、請求項1に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムが暗号化される。また、請求項7に記載の記録媒体には、請求項1に記載の情報処理装置で実行

可能なコードに復号化される暗号文に、プログラムを暗号化したものが記録されている。従って、請求項1に記載の情報処理装置で実行可能な、暗号化されたプログラムの提供が可能となる。

【0141】請求項8に記載の情報処理装置および請求項10に記載の情報処理方法によれば、プログラムが正当なものかどうかを確認され、正当なものであることが確認された場合のみ、プログラムが実行される。また、請求項11に記載の記録媒体には、コンピュータに、プログラムが正当なものかどうかを確認させ、正当なものであることが確認された場合のみ、プログラムを実行させるためのプログラムが記録されている。従って、正当なプログラムのみが実行されるようにすることが可能となる。

【0142】請求項12に記載の情報処理装置および請求項13に記載の情報処理方法によれば、請求項8に記載の情報処理装置において正当なものであると確認されるように、プログラムが処理される。また、請求項14に記載の記録媒体には、請求項8に記載の情報処理装置において正当なものであると確認されるように処理されたプログラムが記録されている。従って、請求項8に記載の情報処理装置で実行可能なように処理されたプログラムの提供が可能となる。

【図面の簡単な説明】

【図1】計算機1の資源と、そこに実装されたJava仮想マシン11の資源との対応関係を示す図である。

【図2】Java仮想マシン11の処理を説明するための図である。

【図3】Java仮想マシン11の処理を説明するための図である。

【図4】Java仮想マシン11の処理を説明するための図である。

【図5】本発明を適用したプログラム提供システムの一実施の構成例を示すブロック図である。

【図6】図5のソフトウェア開発者サーバ31の構成例を示すブロック図である。

【図7】図5のプログラム認証機関サーバ32の構成例を示すブロック図である。

【図8】図5のユーザ端末33の構成例を示すブロック

図である。

【図9】ソフトウェア開発者サーバ31の処理を説明するためのフローチャートである。

【図10】プログラム認証機関サーバ32の処理を説明するためのフローチャートである。

【図11】暗号化／復号化システムの構成例を示すブロック図である。

【図12】プログラム実行システムの第1の機能的構成例を示すブロック図である。

【図13】プログラム実行システムの第2の機能的構成例を示すブロック図である。

【図14】プログラム認証機関サーバ32の処理を説明するためのフローチャートである。

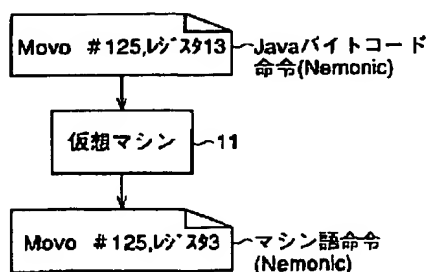
【図15】デジタルサインを用いた暗号化／復号化システムの構成例を示すブロック図である。

【図16】プログラム実行システムの第3の機能的構成例を示すブロック図である。

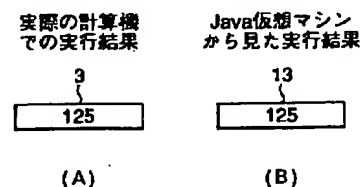
【符号の説明】

1 計算機, 2 中央演算処理装置, 3 レジスタ, 4 メモリ, 11 Java仮想マシン, 13 レジスタ, 14 メモリ, 21 Javaコンパイラ, 31 ソフトウェア開発者サーバ, 32 プログラム認証機関サーバ, 33 ユーザ端末, 34 ネットワーク, 35 記録媒体, 41 CPU, 42 ROM, 43 RAM, 44 入力部, 45 出力部, 46 補助記憶装置, 47 通信制御部, 51 CPU, 52 ROM, 53 RAM, 54 入力部, 55 出力部, 56 補助記憶装置, 57 通信制御部, 61 CPU, 62 ROM, 63 RAM, 64 入力部, 65 出力部, 66 補助記憶装置, 67 通信制御部, 71 暗号化器, 72 復号化器, 81 入力部, 82 復号部, 83 Java仮想マシン, 91 ダイジェスト作成器, 92 暗号化器, 93 復号化器, 94 ダイジェスト作成器, 95 署名確認器, 101 入力部, 102 メッセージダイジェストシステム, 103 署名確認部, 104 仮想マシン入力制御部

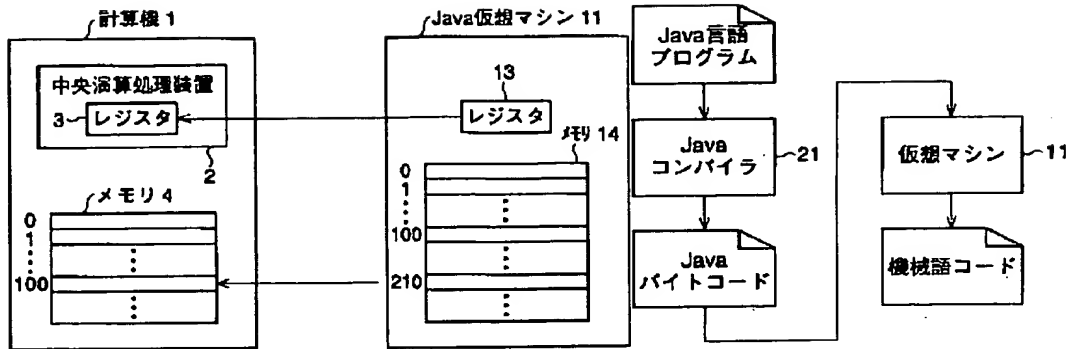
【図3】



【図4】

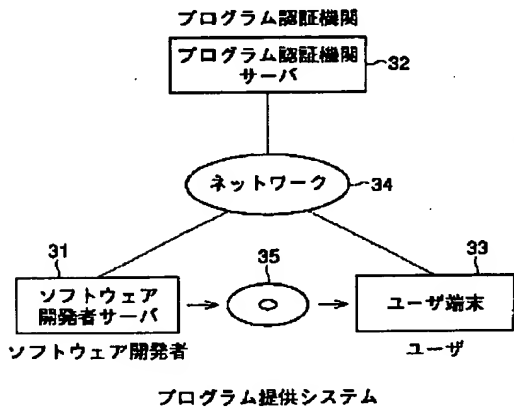


【図1】

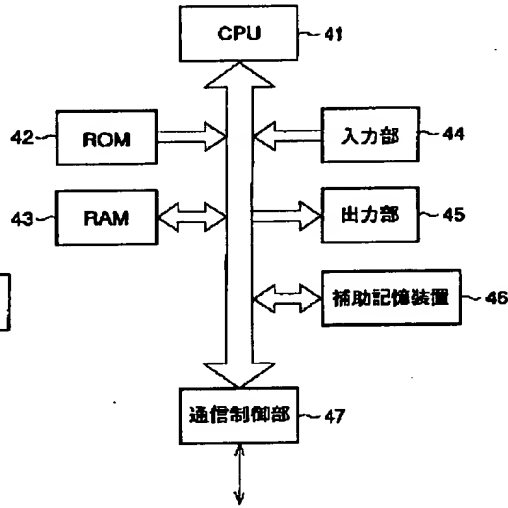


【図2】

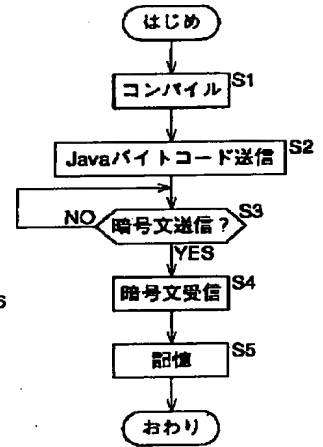
【図5】



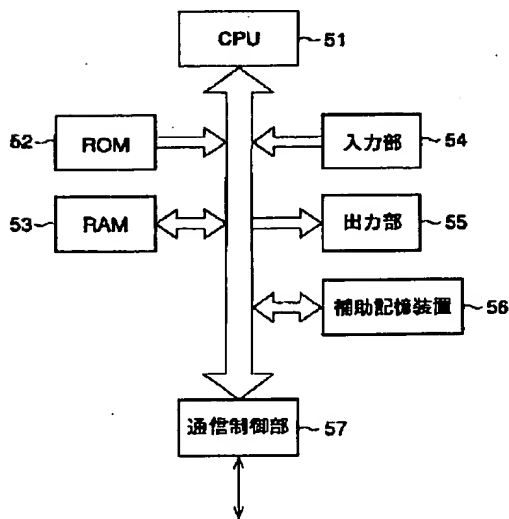
【図6】



【図9】



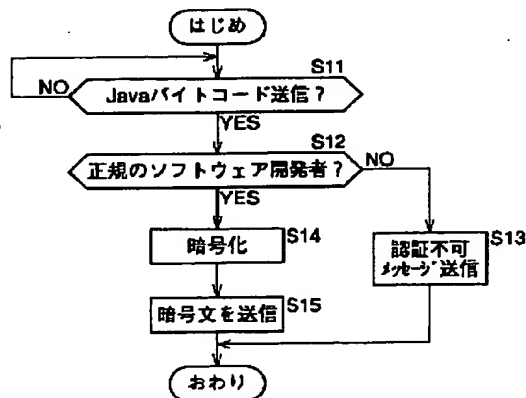
【図7】



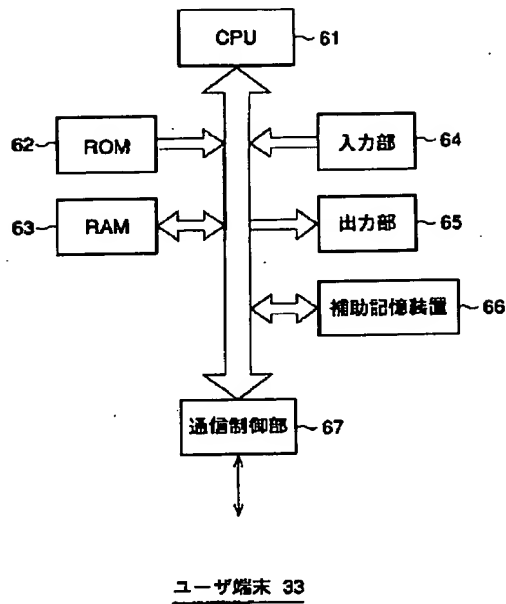
プログラム認証機関サーバ 32

ソフトウェア開発者サーバ 31

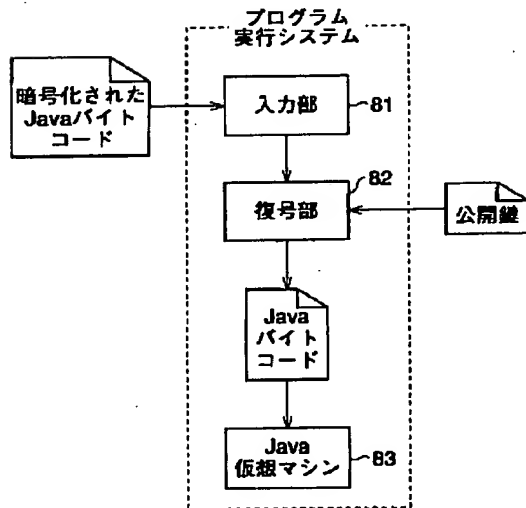
【図10】



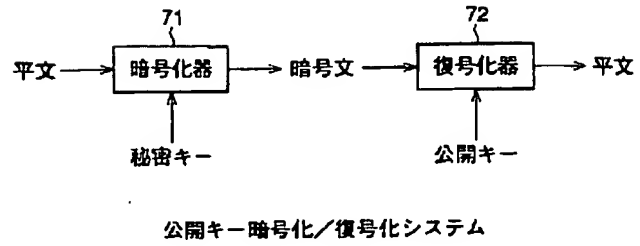
【図8】



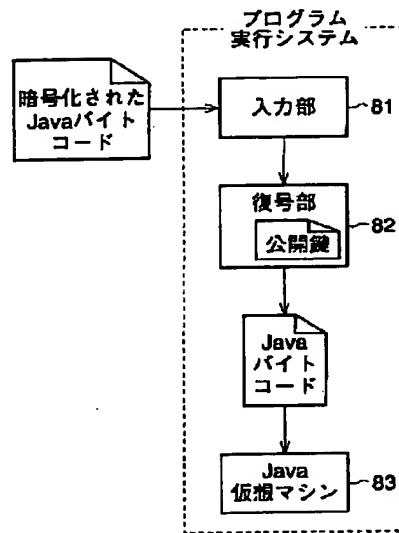
【図12】



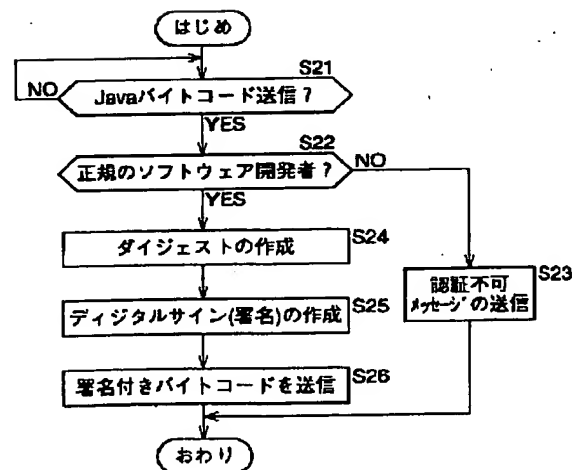
【図11】



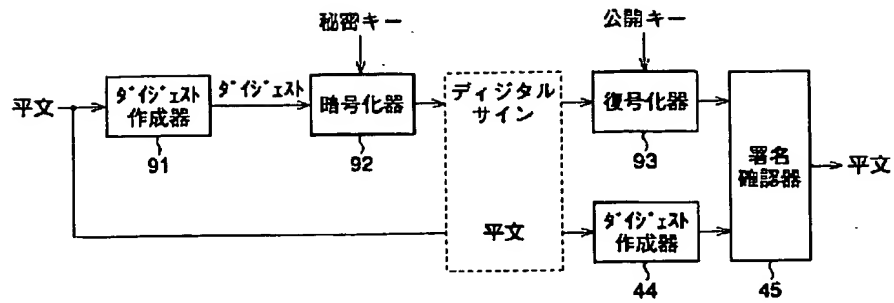
【図13】



【図14】



【図15】



デジタルサインを用いた暗号化／復号化システム

【図16】

